
Formulario de aprobación de curso de posgrado/educación permanente

Asignatura: Profesión en ingeniería de software

Modalidad:

(posgrado, educación permanente o ambas)

Posgrado

Educación permanente

Profesor de la asignatura ¹: Dr. Ing. Diego Vallespir, Profesor Agregado, Instituto de Computación
(título, nombre, grado o cargo, instituto o institución)

Profesor Responsable Local ¹:

(título, nombre, grado, instituto)

Otros docentes de la Facultad: Msc. Lucía Camilloni, Asistente Grado 2, Instituto de Computación
(título, nombre, grado, instituto)

Docentes fuera de Facultad:

(título, nombre, cargo, institución, país)

¹ Agregar CV si el curso se dicta por primera vez.

(Si el profesor de la asignatura no es docente de la Facultad se deberá designar un responsable local)

[Si es curso de posgrado]

Programa(s) de posgrado: Maestría en Informática, Doctorado en Informática

Instituto o unidad: Instituto de computación

Departamento o área: Ingeniería de Software

Horas Presenciales: 16

(se deberán discriminar las horas en el ítem Metodología de enseñanza)

Nº de Créditos: 6

[Exclusivamente para curso de posgrado]

(de acuerdo a la definición de la UdelaR, un crédito equivale a 15 horas de dedicación del estudiante según se detalla en el ítem Metodología de enseñanza)

Público objetivo: Estudiantes de posgrado interesados en temas de la profesión del ingeniero de software.

Cupos: no tiene

(si corresponde, se indicará el número de plazas, mínimo y máximo y los criterios de selección. Asimismo, se adjuntará en nota aparte los fundamentos de los cupos propuestos. Si no existe indicación particular para el cupo máximo, el criterio general será el orden de inscripción, hasta completar el cupo asignado)

Objetivos: Brindar un panorama general sobre la profesión en ingeniería de software discutiendo diversos aspectos de la misma.

Se busca que a través de la realización de este curso el estudiante pueda:

- Conocer y utilizar el cuerpo de conocimiento de la ingeniería de software (SWEBOK) así como saber de sus limitaciones.
- Conocer sobre los aspectos que definen una profesión y en particular la profesión en la ingeniería de software.
- Conocer y diferenciar distintos elementos que conforman a una profesión en ingeniería, tales como: educación profesional inicial, acreditaciones, desarrollo de habilidades, certificaciones, licenciamiento, desarrollo profesional, sociedades profesionales y código de ética. Como ejemplo se discutirán artículos que presentan cómo está hoy en día la Ingeniería de Software con respecto a cada uno de estos elementos.

Conocimientos previos exigidos: Conocimientos de los fundamentos de la ingeniería de software y el desarrollo de software.

Conocimientos previos recomendados:

Metodología de enseñanza:

(comprende una descripción de la metodología de enseñanza y de las horas dedicadas por el estudiante a la asignatura, distribuidas en horas presenciales -de clase práctica, teórico, laboratorio, consulta, etc.- y no presenciales de trabajo personal del estudiante)

Descripción de la metodología:

Clases presenciales obligatorias una vez por semana de 2 horas. Son un total de 8 clases presenciales.

El curso se divide en dos partes. La primera es la parte teórica y la segunda es un trabajo en grupo.

Primera parte

La primera parte del curso se basa en libros y artículos que los estudiantes deberán leer y comprender durante el curso. Los libros y artículos están escritos en su gran mayoría en inglés. Los estudiantes deberán preparar y presentar cada una de las clases a menos de algún tema puntual. Estas clases estarán basadas en el material de lectura. La preparación y presentación de las clases se realizará por parte de grupos de estudiantes conformados para tal fin.

Todas las clases, a excepción de la primera (que será una clase teórica introductoria), se dividirán en 4 partes:

1. Parciales de control de lectura de artículos al comienzo de cada clase.
2. Presentaciones de artículos por grupos de estudiantes.
3. Preguntas sobre los artículos preparadas por grupos de estudiantes.
4. Discusión de los temas presentados en los artículos.

Las clases comenzarán con una prueba parcial individual que buscará evaluar los conocimientos adquiridos por cada estudiante.

Para las clases se asignarán artículos para leer a todos los estudiantes y estos deberán ser presentados por parte de uno o más grupos de estudiantes. Se debe evitar hacer una

presentación resumida del artículo ya que todos los estudiantes lo han leído antes de clase. Entonces, la presentación se deberá enfocar en presentar algún tipo de discusión o análisis sobre el/los artículos que les fueron asignados al grupo.

Además, los estudiantes tendrán que buscar un nuevo artículo, vinculado al tema en cuestión, para presentar al resto de los grupos. Este artículo, a diferencia del asignado por los docentes, al no ser leído por el resto de los estudiantes, debe ser presentado enfocándose en explicar lo que los autores proponen. Obviamente también se puede realizar un análisis crítico.

A su vez, uno o más grupos deberán preparar preguntas sobre los artículos que fueron enviados para leer para discutir en clase.

Las clases también tendrán un espacio de media hora de discusión guiado por los docentes de los temas presentados. Los estudiantes deberán participar en clase discutiendo cada uno de los temas.

Segunda parte

Los estudiantes desarrollarán, con los mismos grupos definidos en la primera parte, un trabajo escrito vinculado a los temas vistos en el curso. El trabajo deberá presentarse en clase. El cuerpo docente propondrá varios temas de los cuales cada grupo elegirá uno.

La dedicación del estudiante se distribuye de la siguiente manera:

- 16 horas de clase presenciales (8 clases de 2 horas.)
- 31 horas de lectura de artículos y estudio individual.
- 17 horas de búsqueda de artículos, preparación de presentación de artículos y discusión de artículos.
- 26 horas en preparación del trabajo grupal.

Detalle de horas:

- Horas de clase (teórico): 14
- Horas de clase (práctico):
- Horas de clase (laboratorio):
- Horas de consulta:
- Horas de evaluación: 2
 - Subtotal de horas presenciales: 16
- Horas de estudio: 48
- Horas de resolución de ejercicios/prácticos:
- Horas proyecto final/monografía: 26
 - Total de horas de dedicación del estudiante: 90

Forma de evaluación:

La evaluación consta de pruebas individuales escritas y presentación de trabajos grupales.

Temario:

1. Introducción
 - 1.1. Definición de la ingeniería de software como disciplina.
 - 1.2. Diferencias entre la ciencia de la computación y la ingeniería de software.
 - 1.3. Qué son las organizaciones internacionales ACM e IEEE-CS.
 - 1.4. Introducción básica a los temas del curso.
2. Cuerpo de conocimiento en ingeniería de software (SWEBOK).
 - 2.1. El cuerpo de conocimiento y su importancia.
 - 2.2. Áreas que conforman el cuerpo de conocimiento.
 - 2.3. Relación entre el cuerpo de conocimiento, la Educación y la Profesión en ingeniería de software.
 - 2.4. Historia del cuerpo de conocimiento.
3. Profesión en ingeniería de software.
 - 3.1. ¿Qué define a una profesión?
 - 3.2. La ingeniería de software como una profesión.
 - 3.3. Ética en ingeniería de software.
 - 3.4. Licenciamiento en ingeniería de software y otras “credenciales”
 - 3.5. Habilidades del ingeniero de software

Bibliografía:

(título del libro-nombre del autor-editorial-ISBN-fecha de edición)

1. Parnas, David (2011). Software Engineering - Missing in Action: A Personal Perspective, IEEE Computer, vol. 44, no. 10, pp. 54-58.
2. Joint Task Force on Computing Curricula IEEE Computer Society Association for Computing Machinery (2015). Software Engineering 2014 - Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, pp. 10-17.
3. The Association for Computing Machinery (ACM); The Association for Information Systems (AIS); The Computer Society (IEEE-CS), (2005). Computing Curricula 2005 - The Overview Report covering undergraduate degree programs in Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering, cap. 2 pp 13-15.
4. Brooks, Frederick (1987). No Silver Bullet Essence and Accidents of Software Engineering. IEEE Computer Society Press, vol. 20, no. 4, pp. 1-5.
5. P. Bourque and R. E. Fairley (2014). Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE Computer Society Press.
6. Camilloni, Lucia; Vallespir, Diego (2014). Sobre profesión y educación en ingeniería de software, Reporte Técnico InCo/PEDECIBA (versión borrador).
7. McConnell, Steve; Tripp Leonard (2013); Software Engineering Professional Practices, en Software Engineering Essentials, Volume II: The Supporting Processes, Richard Hall Thayer and Merlin Dorfman, Eds., 2013, cap. 11.
8. Joint Task Force on Software Engineering Ethics and Professional Practices, IEEE Computer Society and Association for Computing Machinery (1999). Software Engineering Code of Ethics and Professional Practice.
9. Rogerson, Simon (2002). The Software Engineering Code of Ethics and Professional Practice: a case for being proactive, in 26 th Annual International Computer Software and Applications Conference.
10. Bagert, Donald J. (1999). Licensing Software Engineers: The Computer Society Must Continue to Lead, in COMPSAC.
11. Laplante, Phillip A. (2013). An International Perspective on U.S. Licensure of Software Engineers, Technology and Society, vol. 32, no. 1, pp. 28-30.
12. Kruchten, Philippe (2008). Licensing Software Engineers?; IEEE Software 25(6): pp. 35-37.
13. Ellis, Heidi J.C., Steven A (2009). Demurjian and J. Fernando Naveda; Software Engineering: Effective Teaching and Learning Approaches and Practices, IGI Global, cap. 1-17-18.

14. Rivera-Ibarra, J.G.; Rodriguez-Jacobo, J.; Serrano-Vargas, M.A. (2010), Competency Framework for Software Engineers, Software Engineering Education and Training (CSEE&T), 2010 23rd IEEE Conference on , vol., no., pp.33,40, 9-12 March 2010.

15. IEEE CS (2014). The software engineering competency model (SWECOM), draft version.

16. Sedelmaier, Y.; Landes, D. (2012). A research agenda for identifying and developing required competencies in software engineering, Interactive Collaborative Learning (ICL), 2012 15th International Conference on , vol., no., pp.1,5, 26-28 Sept. 2012.

17. Shaw, Mary (2009). Continuing prospects for an engineering discipline of software, IEEE Software, vol. 26, no. 6, pp. 64–67.

18. McConnell, Steve;Tripp Leonard (1999). Professional Software Engineering-Fact or Fiction?, IEEE Software, vol. 16, no. 6, pp. 13-18.

Datos del curso

Fecha de inicio y finalización: Segundo semestre.

Horario y Salón: A confirmar.

Arancel:

[Si la modalidad no corresponde indique "no corresponde". Si el curso contempla otorgar becas, indíquelo]

Arancel para estudiantes inscriptos en la modalidad posgrado:

Arancel para estudiantes inscriptos en la modalidad educación permanente: no corresponde
